SCADA

From Wikipedia, the free encyclopedia.

This article needs to be **cleaned up** to conform to a higher standard of quality.
This article has been tagged since August 2005.
See How to Edit and Style and How-to for help, or this article's talk page.

SCADA (Supervisory Control and Data Acquisition) systems are used in industrial and engineering applications to monitor and control distributed systems from a master location. SCADA is a very broad umbrella that describes solutions across a large variety of industries, including but not limited to the following:

- Electric power generation, transmission and distribution
- Oil and Gas Industry metering and control systems
- Environmental control systems
- Traffic signals
- Water management systems
- Mass transit systems
- Manufacturing systems

The three components of a SCADA system are:

1. Multiple field RTUs, i.e., Remote Terminal Units.
2. Central Control Room with Host Computer(s)
3. Communication infrastructure

The Remote Terminal Unit RTU connects to physical equipment such as switches, pumps, and other devices and monitors and controls these devices.

As the term SCADA implies, the Host computers allow for "supervisory level" control of the remote site. But also "acquire data" from the remote field RTUs. The bulk of the site control is actually performed automatically by the RTU via a local field computer terminal. Host control functions are restricted to basic site over-ride or supervisory level capability.

Data acquisition begins at the RTU level and includes meter readings and equipment statuses that are copied or transferred to the Host as required in a digestible format so that Host control room operators can make appropriate supervisory decisions that may be required to over-ride normal RTU controls. and to al. Due to the sensitivity of the acquired critical data the entire system has restricted access.

While the SCADA human-machine interface (HMI) usually allows operators to view the state of any part of the plant equipment, most operator interaction with the system is driven by alarms. Alarms

are automatically detected abnormal conditions in the plant equipment that require operator attention, and may require operator intervention to keep things running smoothly.

The HMI/SCADA industry was essentially born out of a need for a front-end to a programmable logic controller (PLC). While a PLC does provide automated, pre-programmed control over a process, a PLC is typically a blank box full of devices and does not offer any indication of the health, status, or state of the equipment, nor the ability to readily tap into the program commands. An HMI usually displays sensor information in its physical context, within a graphical depiction of the piping or electrical system in which it resides, allowing the operator to "see what the PLC is doing" to some extent. A sophisticated HMI may also be linked to a database to provide instant trending, diagnostic data, scheduled maintenance procedures, logistic information, detailed schematics for a particular sensor or machine, and expert-system troubleshooting guides. Since about 1998, virtually all major PLC manufacturers have offered integrated HMI/SCADA systems, many of them using open and non-proprietary communications protocols. Numerous specialized third-party HMI/SCADA packages offering built-in compatibility with most major PLC's have also entered the market, allowing mechanical engineers, electrical engineers and technicians to configure HMI's themselves, without the need for a custom-made program written by a software developer.

SCADA master computers typically run on top of a third party operating system. Nearly all SCADA products run on either a UNIX variant or HP OpenVMS, although many vendors are beginning to provide Microsoft Windows as a host operating system option. Initially, more "open" platforms such as Linux were not as widely used due to the highly dynamic development environment and because a SCADA customer that was able to afford the field hardware and devices to be controlled could usually also purchase UNIX or OpenVMS licenses. However, in recent years all SCADA vendors have moved to NT and some also to Linux.

SCADA systems typically implement a distributed database which contains data called points. A point represents a single input or output value monitored or controlled by the system. Points can be either "hard" or "soft". A hard point is representative of an actual input or output connected to the system, while a soft point represents the result of logic and math operations applied to other hard and soft points.

The HMI package for the SCADA system typically includes a drawing program which the operators or system maintenance personnel use to change the way these points are represented in the interface. These representations can be as simple as a on-screen traffic light which represents the state of an actual traffic light in the field, or as complex as a multi-projector display representing the position of all of the elevators in a skyscraper or all of the trains on a railway. The interface is usually 2D and is displayed using the X11 Protocol, although some vendors provide immersive 3D interfaces and support for other display APIs such as Win32 GDI/DirectDraw.

Since the early 1990s the role of SCADA systems in large civil engineering solutions has changed, requiring them to perform more operations automatically. Solutions sold as SCADA also often have Distributed Control System (DCS) components. Use of "smart" RTUs or PLCs, which are capable of autonomously executing simple logic processes without involving the master computer, is

increasing. A functional block programming language, IEC 61131-3, is frequently used to create programs which run on these RTUs and PLCs. Unlike a procedural language such as the C programming language or FORTRAN, IEC 61131-3 has minimal training requirements. This allows SCADA system engineers to perform both the design and implementation of a program to be executed on a RTU or PLC.

For example, instead of relying on operator intervention, or master station automation, RTUs may now be required to operate on their own to control tunnel fires or perform other safety-related tasks. The master station software is required to do more analysis of data before presenting it to operators including historical analysis and analysis associated with particular industry requirements. Safety requirements are now being applied to the system as a whole, and even master station software must meet stringent safety standards for some markets.

For some installations the costs that would result from the control system failing is extremely high. Possibly even lives could be lost. Hardware for SCADA systems is generally ruggedized to withstand temperature, vibration, and voltage extremes, but in these installations reliability is enhanced by having redundant hardware and communications channels. A failing part can be quickly identified and its functionally automatically taken over by backup hardware. A failed part can often be replaced without interrupting the process. The reliabillity of such systems can be calculated statistically and is stated as the mean time to failure, which is a variant of mean time between failures. The calculated mean time to failure of such high reliability systems can be in the centuries.

SCADA systems have traditionally used combinations of radio and direct serial or modem connections to meet communication requirements, although Ethernet and IP over SONET is also frequently used at large sites such as railways and power stations.

This has also come under threat with some customers wanting SCADA data to travel over their pre-established corporate networks, or to share the network with other applications. The legacy of the early low-bandwidth protocols remains, though. SCADA protocols are designed to be very compact and many are designed to send information to the master station only when the master station polls the RTU. Current standard SCADA protocols include Modbus, Conitel, DNP3, IEC 60870-5-101 and RP-570. Many of these protocols now contain extensions to operate over TCP/IP, although it is good security engineering practice to avoid connecting SCADA systems to the Internet so the attack surface is reduced.

The trend is for PLC and HMI/SCADA software to be more "mix-and-match". In the mid 1990's, the typical DAQ I/O manufacturer offered their own proprietary communications protocols over a suitable-distance carrier like RS-485. Towards the late 1990's, the shift towards open communications continued with I/O manufacturers offering support of open message structures like Modicon MODBUS over RS-485, and by 2000 most I/O makers offered completely open interfacing such as Modicon MODBUS over TCP/IP. The primary barriers of Ethernet TCP/IP's entrance into industrial automation (determinism, synchonization, protocol selection, environment suitability) are still a concern to a few extremely specialized applications, but for the vast majority

of HMI/SCADA markets these barriers have been broken.

See also:

- Distributed Control Systems
- Energy Management Systems

Retrieved from "http://en.wikipedia.org/wiki/SCADA"

Categories: Cleanup from August 2005 | Production and manufacturing

---

- This page was last modified 17:45, 25 August 2005.
- All text is available under the terms of the GNU Free Documentation License (see **Copyrights** for details).